WHAT IS CLAIMED IS:

1. A method for keeping two independent busses coherent comprising:

writing data from an Input/Output (I/O) controller to a memory, the I/O controller sending the data to the memory via a first bus connected to a first port of a memory controller and the I/O controller;

sending a tag, from the I/O controller, after the data via the first bus through the first port, the tag being received by the memory controller;

requesting completion status of the data write from the I/O controller by a processing unit, the request being sent to the I/O controller via a second bus connected to a second port of the memory controller and the I/O controller; and

waiting for a tag acknowledgment, by the I/O controller, from the memory controller before providing notification to the processing unit that the data write has completed,

wherein the first bus and the second bus are coherent.

2. The method according to claim 1, comprising writing the data to the memory from one of at least one Direct Memory Access (DMA) controller at the I/O controller.

3. The method according to claim 2, wherein at least one DMA controller comprises a Universal Serial Bus (USB) controller.

4. The method according to claim 2, wherein at least one DMA controller comprises an Integrated Drive Electronics (IDE) controller.

5. The method according to claim 2, further comprising sending the data from the one of at least one DMA controller to a second memory at the I/O controller and then the writing the data to memory.

6. The method according to claim 5, further comprising sending the data, by the I/O controller, from the second memory via the first bus to a third memory at the memory controller and then the writing the data to memory.

7. The method according to claim 6, further comprising sending the tag acknowledge from the memory controller to the one of at least one DMA controller via the third memory, first bus, and second memory.

8. The method according to claim 7, further comprising sending the request for completion status to the one of at least one DMA controller from the processing unit, the request being sent via a fourth memory at the memory controller, the second bus, and a fifth memory located at the I/O controller.

9. The method according to claim 8, further comprising routing the request to an appropriate one of the at least one DMA controller by a crossover unit, the crossover unit operatively connected between each at least one DMA controller and the fifth memory.

10. A system for keeping two independent busses coherent comprising:

at least one memory device;

a memory controller operably connected to the at least one memory device;

a processing unit operably connected to the memory controller; and

an Input/Output (I/O) controller operably connected to the memory controller by a first bus and a second bus, the I/O controller writing data to the at least one memory device via the first bus and the memory controller, the I/O controller sending a tag after the memory write to the memory controller via the first bus, the processing unit requesting status from the I/O controller via the memory controller and the second bus,

wherein the I/O controller waits for a tag acknowledgment from the memory controller before providing notification to the processing unit via the second bus that the data write has completed ensuring that the first bus and the second bus are coherent.

11. The system according to claim 10, further comprising a second memory at the memory controller and a third memory at the I/O controller, the data write sent from the I/O controller to the memory via the third memory, first bus, and second memory.

12. The system according to claim 10, further comprising a fourth memory at the memory controller and a fifth memory at the I/O controller, the processing unit sending the status request to the I/O controller via the fourth memory, second bus, and fifth memory.

13. The system according to claim 12, further comprising at least one Direct Memory Access (DMA) controller at the I/O controller, each at least one DMA controller operatively connected to the third memory, the data write and tag sent from the I/O controller originating at one of the at least one DMA controllers.

14. The system according to claim 13, further comprising a crossover unit operatively connected between the fifth memory and each at least one DMA controller, the crossover unit routing the request to the appropriate at least one DMA controller, the appropriate at least one DMA controller waiting for the tag acknowledgment from the memory controller before providing notification to the processing unit via the crossover unit, fifth memory, second bus, and fourth memory that the data write has completed.

15. The system according to claim 13, wherein at least one DMA controller comprises a Universal Serial Bus (USB) controller.

16. The system according to claim 13, wherein at least one DMA controller comprises an Integrated Drive Electronics (IDE) controller.

17. A memory controller comprising:

a memory interface;

a processing unit interface

at least one first memory device, each at least one first memory device operably connected to the memory interface;

a second memory device operably connected to the processing unit;

at least one first port, one at least one first port operably connected to one at least one first memory; and

a second port operably connected to the second memory,

wherein data to be sent to the memory interface is received through the at least one first port, information from the processing interface is sent to the second port, information to the processing interface is received from the second port.

18. The controller according to claim 17, wherein the at least one first memory and the second memory comprise first-in-first-out (FIFO) devices.

19. The controller according to claim 17, wherein the memory controller comprises an Integrated Circuit (IC).

20. An Input/Output (I/O) controller comprising:

at least one first memory;

at least one Direct Memory Access (DMA) controller, each at least one DMA controller operably connected to one at least one first memory;

at least one first port, each at least one first port operably connected to one at least one first memory;

a second port;

a second memory operably connected to the second port; and

a crossover unit, the crossover unit operably connected between the second memory and each at least one DMA controller,

wherein information received at the second port is routed to the appropriate at least one DMA controller by the crossover unit, the at least one DMA controller sending memory writes via the at least one first memory and the at least one first port.

21. The controller according to claim 20, wherein the at least one first memory comprises a first-in-first-out (FIFO) memory.

22. The controller according to claim 20, wherein the at least one first memory comprises a first-in-first-out (FIFO) memory.

23. The controller according to claim 20, wherein the I/O controller comprises an Integrated Circuit (IC).

24. A system for keeping two independent busses coherent comprising:

at least one memory device;

a memory controller operably connected to the at least one memory device;

at least one processing unit operably connected to the memory controller;

at least one Input/Output (I/O) controller;

at least one first bus, one associated at least one first bus operably connected between one at least one I/O controller and the memory controller;

a second bus operably connected between the memory controller and each at least one I/O controller, each at least one I/O controller writing data to the at least one memory device via the associated at least one first bus and the memory controller, each at least one I/O controller sending a tag after the memory write to the memory controller via the associated first bus, the processing unit requesting status from each at least one I/O controller that initiates the write via the memory controller and the second bus,

wherein each at least one I/O controller waits for a tag acknowledgment from the memory controller before providing notification to the processing unit via the second bus that the data write has completed ensuring that each at least one first bus and the second bus are coherent.

25. The system according to claim 24, further comprising a second memory at the memory controller and a third memory at the I/O controller, the data write sent from the I/O controller to the memory via the third memory, first bus, and second memory.

26. The system according to claim 24, further comprising a fourth memory at the memory controller and a fifth memory at the I/O controller, the processing unit sending the status request to the I/O controller via the fourth memory, second bus, and fifth memory.

27. The system according to claim 26, further comprising at least one Direct Memory Access (DMA) controller at the I/O controller, each at least one DMA controller

operatively connected to the third memory, the data write and tag sent from the I/O controller originating at one of the at least one DMA controllers.

28. The system according to claim 27, further comprising a crossover unit operatively connected between the fifth memory and each at least one DMA controller, the crossover unit routing the request to the appropriate at least one DMA controller, the appropriate at least one DMA controller waiting for the tag acknowledgment from the memory controller before providing notification to the processing unit via the crossover unit, fifth memory, second bus, and fourth memory that the data write has completed.

29. The system according to claim 27, wherein at least one DMA controller comprises a Universal Serial Bus (USB) controller.

30. The system according to claim 27, wherein at least one DMA controller comprises an Integrated Drive Electronics (IDE) controller.